

# 비디오 데이터베이스에서 동작 표현 모델링 및 검색 기법

## (Motion Modeling and Retrieval in Video Databases)

김기병<sup>†</sup> 김명원<sup>\*\*</sup> 김형주<sup>\*\*\*</sup>  
(Ki-Byoung Kim) (Myeong-Wine Kim) (Hyoung-Joo Kim)

**요약** 비디오 데이터베이스의 주된 특징은 공간 데이터(spatial data)와 더불어 시공간 데이터(spatio-temporal data)가 포함되어 있다는 점이다. 내용 기반 비디오 검색의 주대상은 대상물(spatial data)과 동작(spatio-temporal data)임이 알려져있다. 본 논문에서는 중요한 시공간 데이터로서 동작을 모델링하고 이를 검색하기 위한 효율적인 방법을 제시하였다. 동작을 표현하기 위해서, 주기 시퀀스를 이용한 비디오 동작 표현 모델을 제안하였다. 특히 동작의 경우, 동작을 분해하여 이를 주기 시퀀스로 표현할 수 있음을 보이고, 이의 빠른 검색 기법을 제시하였다.

제안된 검색 방법은 모든 속성이 정확히 일치하는 데이터를 검색하는 완전 일치 검색 및, 유사성 측정 에 의한 유사 검색을 지원하며, 원하는 데이터를 실수로 기각하는 경우(false dismissal)가 발생하지 않는다.

**Abstract** A video database contains spatio-temporal data as well as spatial data. It is known that main targets of content-based video retrieval are objects as spatial data and motions as spatio-temporal data. In this paper, we suggest a technique to search motions contained frequently in video databases. We model motions of video data using a sequence and a periodic sequence. We identify a property that the equivalence or similarity of two sequences can be checked easily and simply using permutations of two sequences.

We also suggest a fast retrieval technique for executing queries about periodic sequences on the basis of this property. This method supports not only exact match query, but also similarity query. Furthermore, the proposed method does not produce false dismissals.

### 1. 서론

컴퓨터 하드웨어와 주변 장치의 급속한 발전으로 컴퓨터를 이용하여 비디오 데이터를 저장하거나 검색하는 일이 점차로 보편화되고 있다. 비디오 데이터는 그 데이터 양이 매우 방대하므로 비디오 검색의 중요성이 점차로 부각되고 있다. 또한, 비디오 검색을 위해서 비디오 데이터가 전달하고자 하는 의미를 간결하고 함축적으로 표현할 필요가 있다.

비디오가 포함하고 있는 데이터는 각 이미지 프레임

에 의해 표현되는 공간 데이터(spatial data)와 시간의 추이에 따라 연속된 이미지 프레임을 통하여 의미를 표현하는 시공간 데이터(spatio-temporal data)의 두가지이다. 비디오 데이터를 각 프레임 별로 구분하여 보면, 각각의 프레임은 공간 데이터로 볼 수 있고, 전체 비디오 데이터는 각각의 이미지 데이터를 시간의 흐름에 따라 배열해 놓은 시공간 데이터로 볼 수 있다. 내용기반 검색에서 관심을 가지는 공간 데이터의 대표적인 예로는 비디오나 이미지에 나타나 있는 대상물을 들 수 있고, 시공간 데이터의 대표적인 예로는 대상 객체의 동작이나 궤적을 들 수 있다. 이에 따라 내용기반 비디오 검색의 주대상은 공간 데이터로서 대상물과 시공간 데이터로서 동작임이 제시되어졌다[1, 6].

공간 데이터를 표현하고 검색하기 위한 연구는 디지털 이미지 프로세싱이나 내용 기반 이미지 검색 분야에

<sup>†</sup> 종신회원 : 한국휴렛팩커드(주) 컨설팅사업본부 전자상거래팀 과장

<sup>\*\*</sup> 비 회 원 : 숭실대학교 컴퓨터학부 교수

<sup>\*\*\*</sup> 종신회원 : 서울대학교 컴퓨터공학과 교수

논문접수 : 1996년 1월 16일

심사완료 : 1997년 10월 17일

서 많이 진행되어져 왔다[4,10]. 시간적인 특징을 표현하기 위한 연구는 멀티미디어 데이터의 저작이나 합성에 관한 연구를 중심으로 행해졌다[5,17]. 그러나, 시간적인 특징이나 시공간적인 특징을 검색하기 위한 연구는 미비하다.

본 논문에서는 비디오 데이터가 포함하고 있는 의미 데이터 중 시공간 데이터를 모델링하고 검색하는데 관심을 두었으며, 그 예로서 동작 데이터에 대해 시퀀스를 이용하여 모델링하고 내용 기반 검색을 적용하여 보았다.

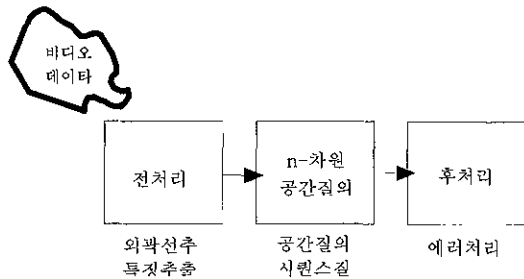


그림 1 멀티미디어 데이터의 일반적인 처리과정

이미지 데이터와 같은 공간 데이터는 그림 1과 같은 단계를 거치면서 검색이 이루어진다. 이미지가 포함하고 있는 대상물의 의미를 표현하기 위해서는 원래의 데이터로부터 추출한 특징 벡터(feature vector)가 주로 사용되므로 공간 데이터 검색 문제는 벡터의 검색 문제로 변환될 수 있다. 즉, 벡터를 이용하면 멀티미디어 데이터가 가지고 있는 공간적인 의미는 쉽게 표현된다. 그러나 시간적 데이터의 표현은 단순히 벡터만을 이용하여 표현하기는 어렵다

그래프[5]나 Petri-net[13]을 이용한 시공간 데이터의 표현은 비디오 데이터가 가지고 있는 시공간적인 데이터를 표현할 수 있다. 이 때, 비디오 데이터가 가지는 시공간 데이터를 그래프에 기반한 방법으로 표현할 경우, 비디오 검색 문제는 그래프 비교 문제로 변환된다. 그러나 그래프에 기반한 비교는 그 연산의 복잡도가 NP-complete이므로, 이를 검색에 사용하는 것은 적절하지 못하다[9]

여기서 멀티미디어 데이터의 저작이나 합성에서 필요한 동기화와 같은 표현은 이미 녹화된 비디오 데이터를 표현하는 경우, 비디오에 포함된 의미가 이미 정해진 형태이므로, 비디오 데이터를 기술하기 위해 별도의 동기화 표현이 필요없음을 알 수 있다. 그러므로 본 논문에서

서는 비디오 데이터와 같이 데이터베이스에 미리 저장되어 있는 데이터가 가지는 시공간 데이터의 의미를 표현하기 위해 시퀀스가 더욱 적절하다. 이에 따라 비디오 스트림에 대한 검색 문제는 시퀀스 검색이나 문자열 검색 문제로 변환될 수 있다. 문자열을 이용한 검색의 경우, 원 문자열의 길이가  $M$ , 대상 문자열의 길이가  $N$  이라면, 최대  $O(M+N)$ 의 복잡도를 가지는 문자열 일치 알고리즘이 존재하므로, 효율적인 시공간 데이터 검색의 기반이 된다[14].

시퀀스를 이용하여 동작을 표현할 경우, 임의의 시퀀스 내에 부 시퀀스가 주기적으로 반복하여 나타나거나, 부 시퀀스의 일부가 중복하여 나타나는 경우가 많다. 특히 이러한 주기성은 동작을 수행하는 객체를 팔이나 다리 등의 구성 요소로 분할하여 모델링할 때 더욱 쉽게 발견된다. 이러한 중복성을 이용하면 시퀀스의 검색을 보다 효율적으로 행할 수 있는데, 이러한 성질을 비디오 검색에 적용하고자 하였다.

본 논문에서는 이를 위해 시퀀스와 주기 시퀀스에 기반한 비디오 시퀀스 모델을 제안하고 중복이나 주기가 있는 시공간 데이터의 빠른 검색 기법을 제시하였다.

이는 비디오 데이터베이스로부터 내용 기반 동작 검색이 다항식(polynomial) 시간의 복잡도를 가지는 비교를 통하여 수행할 수 있는 기반이 된다. 동작을 표현하는 시퀀스 데이터는 주기를 가지거나, 중복된 부 시퀀스를 여러번 포함하는 경우가 많은데, 본 논문에서는 이러한 성질을 이용하여 빠른 검색 방법을 제시하였다. 이를 이용하여 검색을 수행하면, 일치 검색이나, 유사성 검색을 효율적으로 수행할 수 있으며, 잘못된 기각(false dismissal)을 하지 않는다는 것을 보였다. 또한, 이 모델을 기반으로 하여 내용 기반 비디오 검색을 수행하는 원형 시스템(prototype)을 구현하고, 그 검색의 결과를 보였다.

본 논문의 구성은 다음과 같다. 2절에서는 본 연구와 관련하여 내용기반 비디오 검색과 시퀀스 검색에 대한 연구들을 설명하였다. 3절에서는 내용기반 비디오 검색을 위한 비디오 시퀀스 모델과 주기 시퀀스의 성질을 보였다. 4절에서는 제시된 모델에 근거하여, 비디오 스트림과 같은 시퀀스 데이터에 대해 빠른 검색을 수행하기 위한 방법을 제시하였다. 5절에서는 시퀀스 모델에 기반한 내용기반 비디오 검색시스템을 구현하고 검색 결과를 보였다. 끝으로 6절에서는 본 논문에서 제시한 비디오 시퀀스 모델 및 검색 기법의 특징을 요약하고, 앞으로의 개선점에 대하여 이야기하였다.

## 2. 관련 연구

비디오 데이터의 자유로운 표현 능력으로 인하여 비디오 데이터는 점차 널리 사용되고 있다. 이에 따라 비디오 데이터를 컴퓨터를 이용하여 저장하거나 검색하기 위한 필요성도 점차로 증가하고 있다.

Day, Dagtas와 Iino는 [5]에서 그래프에 기반한 비디오 표현 모델을 제시하였다. 그러나 그래프를 이용한 표현 방법은 검색 연산의 복잡도가 크므로 검색에 적용 시 비 효율적일 수 있다.

Dimitrova와 Golshani는 [6]에서 비디오 데이터로부터 대상물의 궤적을 추적하여 이동 경로를 표현하고 검색하기 위한 방법을 제시하였다. 여기에서는 내용기반 비디오 검색의 대상은 대상 객체와 대상 객체의 움직임의 두가지 정보가 대표적임을 제시하였으며 이를 OMV(Object-Motion-Video)모델로 표현하고 질의 방법을 제시하였다. 그러나, 이 논문에서는 대상 객체의 추출은 가정하였으며, 대상 객체의 동작의 경우도 객체의 이동에 따른 궤적만을 표현하고 검색하였으므로, 내용 기반 비디오 검색 시스템으로 확장하기 위해서는 객체의 추출이나, 객체의 구체적인 동작에 대한 표현 및 검색의 보완이 필요하다. 또한, 얻어진 객체의 궤적의 시퀀스가 길어질 경우에 대한 고려가 미비하다.

시퀀스는 시간에 종속적인 개념을 표현하는데 유용하므로, 시퀀스는 주로 시간에 종속적인 의미를 표현하거나 비교하는데 많이 사용되었으며, 비디오 데이터 검색 외에도 멀티미디어 스트림이나, 관측 정보 등은 시퀀스를 이용하여 쉽게 모델링할 수 있다. 이러한 시퀀스 데이터에서 동일한 시퀀스나 유사한 시퀀스를 찾아내는 것은 여러 분야에서 시도되어 왔고, 특히 데이터 마이닝이나 비정형 데이터베이스 질의에서 많은 연구가 행해지고 있다[2,3,7,8,12].

Agrawal, Lin, Sawhney와 Shim은 [3]에서 시퀀스를 비교하여 검색을 행하는 연구를 소개하였는데, 일반적으로 전체 시퀀스를 비교하는 것이 어려우므로 시퀀스의 일부를 윈도우로 지정하여, 윈도우에 대한 검색을 수행한 후, 윈도우를 확장하여 시퀀스의 검색을 수행하였다. 이러한 기법들은 일반적인 시퀀스 검색으로서, 시퀀스의 각각의 값을 비교하는 방법으로 검색을 수행하거나, 변환을 통해 얻어진 값을 이용한 검색을 수행하게 된다. 이러한 방법은 일반적인 시퀀스에 대해서 효과적인 검색 방법이 된다. 그러나 여기에서는 시퀀스로부터 적절한 규칙이나 주기를 발견할 수 있는 경우에는 이러한 정보를 효율적으로 이용할 수 없게 된다.

일반적인 시퀀스 데이터는 그 길이가 크므로 검색을 위하여 인덱스를 사용하는 것이 쉽지 않다. Agrawal, Faloutsos, Swami는 [2]에서 공간 데이터 검색에 주로 사용되는 멀티키를 시퀀스 검색에 적용하기 위해 차원을 줄일 수 있는 방법을 제시하였으며 Faloutsos와 Lin은 [7]에서 Fastmap 기법을 제안하여 시퀀스의 차원을 줄인 후, 그 결과에 대해 기존에 제안된 멀티키 인덱스를 이용하여 효율적인 검색을 수행할 수 있는 방법을 제시하였다. 그러나 이러한 방법은 차원이 낮은 데이터의 시퀀스에 대해서는 적절하게 적용할 수 있으나 벡터의 시퀀스와 같이 차원이 높은 데이터에 대해서는 변환의 결과에 대해서도 동일한 차원의 데이터가 얻어지므로 효율적인 인덱스의 이용이 어렵게 된다.

Seitz와 Dyer는 [15]에서 시퀀스의 특수한 성질인 주기를 affine-invariant로 획득하는 방법을 제안하였다. 여기에서는 대상물의 3차원 좌표를 얻어 이를 순수한 주기를 가지는 동작 벡터로부터 affine-transform을 적용하여 주어진 동작을 표현할 수 있으면, 이의 주기를 모델링하는 방법을 사용한다. 그러나, 이는 처리할 데이터의 양이 비교적 크므로 데이터베이스의 검색 연산으로 사용하기에는 비효율적이다.

Leung과 Yang은 [11]에서 사람의 동작을 추출된 외곽선을 이용하여 표현하는 방법을 제시하였고, Siddiqu와 Kimia는 [16]에서 이미지로부터 대상물의 불변 속성(invariant)을 추출하기 위해 대상 객체의 오목한 부분(neck)을 이용하기 위한 방법을 제시하였다. 본 논문에서는 대상 객체의 동작을 표현하기 위해 이 두 논문에서 제시한 방법으로 특징 벡터를 추출하고 이를 시퀀스로 확장한 후 검색에 사용하였다.

## 3. 비디오 시퀀스 모델

본 절에서는 비디오 데이터를 효율적으로 표현하고 검색할 수 있는 비디오 시퀀스 모델을 제안하고, 그 검색 방법을 보인다.

비디오 DBMS 내에서 비디오로부터 추출된 하나의 이미지 프레임을 표현하기 위해 이미지 분석 기법을 적용하여 이로부터 특징 벡터  $V$ 를 추출하는 것이 일반적이다. 이때 추출된 벡터  $V$ 는

$$(i_1, i_2, i_3, \dots, i_n), i_n \in I, I \text{는 실수의 집합.}$$

과 같은 형태를 가진다. 비디오 데이터일 경우 이러한 이미지가 연속해서 나오게 되므로 비디오 데이터를 표현하기 위해서는 그림 2와 같은 특징 벡터의 시퀀스가

필요하게 된다.

정의 3.1 특징 벡터의 시퀀스  $s$ 는  $\langle v \rangle$ 로 정의된다. 단,  $\langle v \rangle$ 는 동일한 스키마에 속하는 특징 벡터이다. 시퀀스  $\langle v \rangle$ 는  $v_1, v_2, v_3, \dots, v_m$ 을 나타낸다.

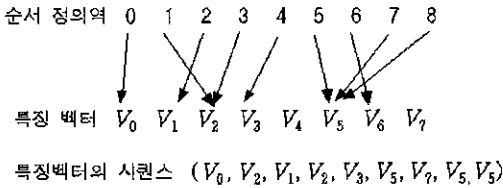


그림 2 특징 벡터의 시퀀스

정의 3.2 주기 시퀀스  $s_p$ 는 순서쌍

$$(\langle v \rangle, t, p)$$

로 정의된다. 단,

- $\langle v \rangle$ 는 특징 시퀀스에서 중복하는 시퀀스의 한 주기의 시퀀스.
- $t \in T$ 는 시퀀스를 구성하는데 사용된 특징 벡터의 종류.  $T$ 는 특징 벡터 타입의 집합.
- $p \in I$ 는 주기 시퀀스의 주기,  $I$ 는 정수의 집합.

특정 시퀀스에 동일한 부분이나 그 차이가 임계값을 넘지 않는 유사한 시퀀스가 반복될 때, 이를 한 주기의 시퀀스와 주기를 이용하여 표현한다. 어떤 시퀀스가 주기를 가질 경우, 그림 3과 같이 시퀀스에 동일한 부분이 일정한 주기를 두고 반복적으로 나타나게 된다. 이를 전부 유지하는 것은 데이터의 중복이 생기므로 주기  $p$  동안의 부시퀀스만을 얻어 반복되는 시퀀스의 내용을 한번만 유지한다. 동일한 시퀀스가 반복되므로, 한 주기의 부시퀀스만을 추출하기 위해, 임의의 시작 위치로부터 주기  $p$  만큼의 부 시퀀스를  $\langle v \rangle$ 로 표현한다.

이제 시퀀스에 표현되어 있는 중복을 검색에 이용하기 위해 중복을 효율적으로 다룰 수 있는 방법을 제시한다.

정의 3.3 순열은 집합이나 시퀀스의 원소를 재배열하는 함수로 정의된다. 이는 임의의 집합  $A$ 에서 집합  $A$ 로의 전단사 함수이다.

순열은 집합의 원소에 대해 임의의 순서가 있다면, 이의 순서를 재배열하는 효과가 있다. 순열(permu-

tation)은 다음과 같은 대수 기호(algebraic notation)로 표현한다.

$$\sigma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 4 & 2 & 5 & 3 & 1 \end{pmatrix}$$

즉 순열  $\sigma$ 는  $\sigma(1)=4, \sigma(2)=2, \sigma(3)=5, \sigma(4)=3$ , 그리고  $\sigma(5)=1$ 을 나타낸다. 일반적으로 대수 식에서의 첫 번째 행은 순서 정의역(ordering domain)의 원래의 순서를 나타내므로, 본 논문에서는 이를 줄여서, 두 번째 행만을 가지고 순열을 표현하도록 하겠다 즉 위의 순열은

$$\sigma = (4 \ 2 \ 5 \ 3 \ 1)$$

로 나타낼 수 있다. 집합  $S$ 의 원소의 갯수가  $n$ 일 경우 순열의 총 갯수는  $n!$ 개가 존재한다.

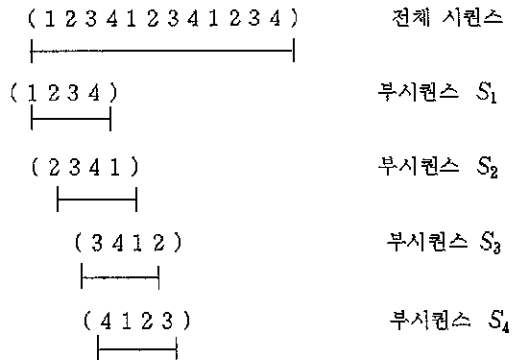


그림 3 주기를 가지는 시퀀스와 부시퀀스

정의 3.4 (순환 순열, 순환 시퀀스, 순환 일치) 크기가  $n$ 인 임의의 시퀀스  $seq_1$ 에 대해, 순열  $\rho$ 를 적용시킬 때,  $seq_1$ 과  $\rho \cdot seq_1$ 의 원소들이 그림 3과 같이 원래의 순서를 유지하면서, 순환하는 모양을 가질 때, 이러한 순열  $\rho$ 를 순환 순열(rotational permutation)이라 정의한다.

임의의 시퀀스  $seq_1$ 에 대해  $seq_2 = \rho \cdot seq_1$ 이고,  $\rho$ 가 순환 순열이라면,  $seq_2$ 를  $seq_1$ 의 순환 시퀀스라 정의한다.

임의의 시퀀스  $seq_1, seq_2$ 에 대해,  $seq_1 = \rho \cdot seq_2$ 를 만족하는 순환 순열  $\rho$ 가 존재하면,  $seq_1$ 과  $seq_2$ 는 순환 일치(rotational equivalence)라고 정의한다.

임의의 집합  $S$ 의 순서 시퀀스에 대해, 이 시퀀스를

순환시키는 순환 순열은 원소의 갯수가  $n$ 개이면,  $n$ 개 존재한다. 동일한 주기 시퀀스를 나타내는 여러가지 표현이 그림 3에 나타나 있다.

성질 3.1 순환 일치에 성질 시퀀스  $seq_1$ 과 순환 시퀀스  $seq_2 = \rho \cdot seq_1$ 에 대해,  $seq_1$ 이 등차 수열이면,  $(seq_1 - seq_2) \bmod n = c \cdot v_n$ 이다. 단  $v_n$ 는  $seq_1, seq_2$ 와 size가 같고 모든 원소가 1인 단위 벡터이며 mod는 나머지 연산이다.

증명:

$seq_1$ 은 등차 수열이므로,

$$seq_1 = (v_1, v_1 + d, v_1 + 2d, v_1 + 3d, \dots, v_1 + (n-1)d) \text{이다.}$$

단,  $v_1$ 은 초항,  $d$ 는 공차,  $n$ 은 시퀀스의 갯수이다.

$$seq_2 = \rho_i \cdot seq_1$$

$$seq_2 = (v_1 + (n-t)d, v_1 + (n-t+1)d, \dots, v_1 + (n-1-t)d)$$

이다.

단,  $\rho_i$ 는  $i$ 만큼 순환시키는 순환 순열. 즉,

$$\rho_i = (t, t+1, t+2, \dots, t-1) \text{이다.}$$

이 때, 두 시퀀스의 차를  $seq_3$ 이라고 하면,

$$\begin{aligned} seq_3 &= (seq_1 - seq_2) \bmod n \\ &= ((t-n)d, (t-n)d, \dots, td) \bmod n \\ &= (td, td, td, \dots, td) \\ &= td \cdot (1, 1, 1, \dots, 1) \quad \square \end{aligned}$$

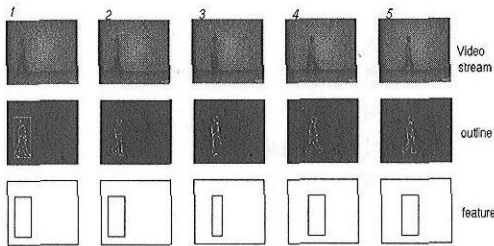


그림 4 MPEG 비디오로부터 특정 동작에 대한 비디오 시퀀스의 추출

이제 시퀀스를 이용하여 이미지 프레임들의 시퀀스로서 비디오 스트림을 정의한다.

정의 3.5 비디오 구간(shot)  $o$ 는 순서쌍  $(id_o, S_p, f)$ 로 정의된다. 단,

- $id_o$ : 하나의 비디오 구간을 나타내는 비디오 구

간 식별자.

- $S_p$ : 주기 시퀀스  $s_p$ 의 집합. 하나의 비디오 구간에 포함된 여러가지 주기 시퀀스들의 집합을 나타낸다.
- $f$ : 샘플링 속도. 매초마다  $f$  개의 프레임이 샘플링되었음을 나타낸다.

정의 3.6 비디오 데이터베이스  $D$ 는 순서쌍  $(id_v, O)$ 로 정의된다. 단

- $id_v$ : 하나의 비디오를 나타내는 비디오의 식별자.
- $O$ : 비디오 구간  $o$ 의 집합.

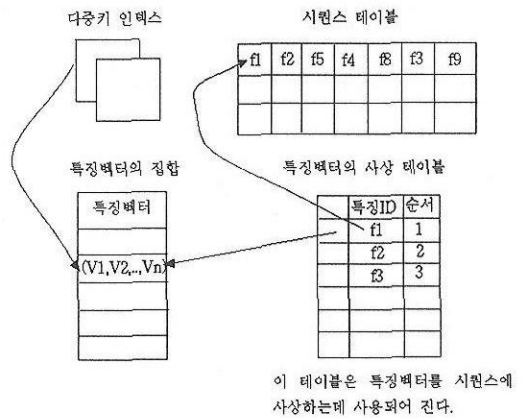


그림 5 동작 검색을 위한 특징 벡터, 시퀀스 및 사상 함수

## 4. 내용 분석 및 검색

### 4.1 내용 분석

시퀀스 검색 및 주기 시퀀스 검색을 이용하여 내용기반 비디오 검색을 수행하기 위해서는, 주어진 비디오 데이터를 분석하여 시퀀스를 얻고, 이로부터 주기를 찾아내는 작업은 질의의 전처리 단계에서 행해야한다.

내용 분석을 위해서 비디오에 포함된 동작을 제한적으로 가정하였다. 분석될 비디오는 고정된 카메라로 사람의 여러가지 동작을 녹화하여 이용하였다. 먼저 녹화된 비디오로부터 MPEG 비디오로 변환한 후, 일정한 간격으로 이미지를 추출하였다. 그림 4와 같이 추출된 이미지에 대해 Sobel 알고리즘을 이용하여 외곽선을 추출하였다. 추출된 데이터로부터 최소사각형을 추출한 후, 각 최소 사각형을 정규화하여 평행이동이나 크기 변

환과 같은 변환에 무관한 불변속성(invariant)이 되도록 하였다. 얻어진 데이터를 순서에 따라 늘어놓으면 비디오에 포함된 동작의 정규화된 특징 벡터의 시퀀스가 된다. 이 때 얻어진 특징벡터들은 대상물의 특징점들의 좌표값이 되는데, 이는 대상물이 움직일 때, 속도등에 영향을 많이 받으며, 특징점의 일부가 다른 물체나 대상물의 일부에 의해 가릴 경우 얻어지지 않을 수 있다. 이런 문제를 해결하기 위해 특징 벡터의 시퀀스를 만들 때, 연속된 특징 벡터로부터 일련하는 두개의 벡터의 차 ( $s_i = v_i - v_{i-1}$ ,  $s_i$ 는 새로운 특징 벡터,  $v_i, v_{i-1}$ 은 원 특징 벡터들)를 이용하여 특징 시퀀스를 작성한다. 이 경우 새로운 특징 벡터에는 동작의 속도가 반영되며, 특징점의 이동에 따른 속도와 가림에 의해 특징 점이 얻어지지 않는 경우, 전후의 값을 이용한 보간(interpolation)을 통하여 처리가 가능하게 된다. 그림 5에서 보여지는 것과 같이 각 시퀀스의 원소들은 테이블에 저장된 후 유사 매트릭을 이용하여 분류되고 해싱을 통하여 인덱스를 얻게 된다. 이를 통하여 시퀀스의 제베치를 인덱스를 이용하여 할 수 있으며 검색 시에 인덱스만을 이용하면 주어진 시퀀스들을 등치수열로 가정할 수 있게 된다. 얻어진 시퀀스 데이터로부터 주기를 가지는 시퀀스를 찾아 서로 구별되는 동작으로 분류한다. 분석된 시퀀스는 그림 6과 같다 이러한 주기성은 대상물을 팔, 다리 등의 구성 요소로 분해할 경우, 더욱 명확하게 나타난다.

동작	특징 시퀀스	주기시퀀스
walk	1,1,2,1,1,2,1,1,2,1,1	(1,1,2) 주기=3
run	1,2,1,2,1,2,1,2,1,2,2	(1,2) 주기=2
jump	1,14,15,1,14,15,1,14	(1,14) 주기=2
raise hand	2,4,5,6,7,6,5,4,2	(2,4,5,6,7,6,5,4,2) 주기=9
bow 1	2,8,9,10,11,12,13,12,11,10,9,8,2	(2,8,9,10,11,12,13,12,11,10,9,8,2) 주기=13
bow 2	2,8,9,10,11,12,13,12,11,10,9,8,2	(2,8,9,10,11,12,13,12,11,10,9,8,2) 주기=13

o 위의 숫자들은 비디오에서 추출된 특징 벡터를 테이블에 저장한 후 이의 인덱스를 이용하여 간략화 표현한 것이며, 같은 숫자는 임계치를 넘지않는 범위 내에서 동일한 특징 벡터로 분류된 것들을 나타낸다.

그림 6 추출된 특징 시퀀스와 주기 시퀀스의 예

주어진 시퀀스로부터 반복하는 부시퀀스를 찾아내기 위해 본 논문에서는 다음과 같은 알고리즘을 이용하였다.

```
Algorithm 1 find_period(string str)
1  i = strien(str)
```

```
2  for i = 1 to t/2
3      for j=k×i to i+k·i
4          if str[j] = str[j+i] then
5              match = match + 1
6  ratio = match / (i·k)
7  if ratio > Threshold then
8      accept i as a dominating period of str
```

임의의 시퀀스에서 주기를 찾는 알고리즘은 다음의 빠른 푸리에 변환(FFT)을 이용하여 시간 정의역(time domain)을 주파수 정의역(frequency domain)으로 변환한 후 여기서 유의한 계수(coefficient)를 사용하여 찾을 수 있다. 일반적으로,  $|X_f|$ 의 주기는 시간 정의역의 주기의 역수인 관계가 성립한다.

$$X_f = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} x_t \exp(-j2\pi f t / n), f=0,1,2, \dots, n-1$$

단,  $j = \sqrt{-1}$ 이며,  $x_t, t=0,1, \dots, n$ 은 임의의 벡터이다.

#### 4.2 검색

시퀀스 검색에서 가능한 검색 종류를 본 논문에서는 다음의 4가지로 구분하고, 각각의 경우에 대해 검색 방법을 제시하였다. 검색을 위해서 시퀀스가 가지는 중복성을 최대한 이용하고자 하였는데, 특히 주기를 가지는 경우 주기를 검색에 이용하여 빠른 검색이 가능하도록 하였다.

- 완전 일치 검색
  1. 완전 일치
  2. 순환 일치
- 유사 검색
  - 1 일치 주기 검색
  2. 비일치 주기 검색

#### 4.3 완전 일치 검색(exact match)

두 시퀀스  $seq_1, seq_2$ 에 대해 시퀀스를 이루는 원소들의 집합  $S_1, S_2$ 가 같고, 주기  $T$ 가 같으며, 두 집합의 원소들의 시퀀스가 일치하면 두 시퀀스는 서로 완전 일치라 한다. 두 시퀀스  $seq_1, seq_2$ 에 대해, 시퀀스를 이루는 두 집합이 일치하고 주기  $T$ 가 같으며, 순환 순열이 존재하면, 정의 3.4에 의해 이 두 시퀀스를 순환 일치라고 한다. 주기 시퀀스에서는 순환 일치하는 시퀀스는 우리가 탐색하고자하는 시퀀스와 같은 시퀀스임을 알 수 있다. 우리는 임의의 주기 시퀀스 간의 일치 검색 문제를 해결하기 위해, 주기 시퀀스 일치 문제를 다음

두개의 부분제(subproblem)로 분할하였다. 이를 위해 두 시퀀스는 알고리즘 1을 이용하여 주기를 알아낸 후, 일단 순열을 이용한 표현으로 변환된다. 기준 시퀀스에는 동차수열에 의한 순서가 부여되고, 비교 대상 시퀀스는 기준 시퀀스에 대한 순열이 할당된다. 이를 위해 기준 시퀀스의 테이블을 구성하고, 해싱 기법을 이용하면 대상 시퀀스가 기준 시퀀스의 몇번째 원소에 대응하는지를 쉽게 알 수 있다.

Step 1 시퀀스를 구성하는 원소들의 일치 확인

- \* 각 집합의 크기를 확인.
- \* 각 집합의 원소를 이용한 차이(distance metric)를 측정<sup>1)</sup>
- \* 각 원소의 포함 관계를 확인

서로 다른 두 시퀀스를 구성하는 원소들이 같은지의 검색은 집합의 동일성 검색과 같다. 즉 주어진 두개의 시퀀스를 구성하는 원소들이 동일한지 확인한다. 각 원소들은 집합에서는 순서를 고려하지 않으므로, 집합에 속한 원소들로부터  $L_p$  메트릭을 구하면, 같은 원소를 가진 집합은 항상 동일한 값이 얻어진다 즉 동일한 원소를 가질 경우, 거리 메트릭의 값은 항상 같은 값이 된다. 단 역은 성립하지 않는다 집합의 메트릭에 의하면 어떤 시퀀스의 원소가 주어진 시퀀스의 원소들과 하나도 일치하지 않으면 이 경우, 메트릭의 값에 충분히 큰 값을 할당한다. 이에 의해, 이 시퀀스는 검색의 후보로 선택되지 않게 된다.

Step 2 순열 확인

- \*  $v_{res} = (v_1 - v_2) \bmod n$ 를 계산한다.
- \* 결과가  $c \cdot v_n$ 인지 확인한다. (단,  $c$ 는 상수,  $v_n$ 는 단위 벡터)

동일한 원소를 가지는 집합이 선택되면, 임의의 집합 S에 대해, 모든 원소의 순서에 따라 해시 테이블을 구성하고 구성된 테이블에 따라, 모든 원소의 인덱스를 구한다. 이러한 인덱스에 대해 순환 일치의 성질을 적용하면, 주어진 두개의 시퀀스에 대해 순환 일치를 확인 할 수 있다.

4.4 유사 검색

임의의 두 시퀀스가 시간의 추이에 대해 충분히 많은 시간에 대한 일치 쌍(time-ordered pair)을 가지면 이

두 시퀀스를 유사하다고 말한다[3]. 주기 시퀀스에서는, 주기 시퀀스를 이루는 원소들에 대해 충분히 많은 원소가 일치하고, 일치한 원소들이 시간에 대한 순서가 같으면, 이를 유사하다고 하자. 즉 두 주기 시퀀스가 한 주기 내에서 충분히 많은 시간에 대한 일치 쌍(time-ordered pair)를 가지면, 이를 유사하다고 말한다.

이 때 임의의 두 시퀀스에 대해, 하나의 시퀀스에 대한, 다른 시퀀스의 순열을 구한 후, 순열에 대한 순환 일치 성질을 확인한다. 이 때, 순환 일치이면, 두 시퀀스를 유사하다고 말할 수 있다 임의의 시퀀스에 속한 원소들을 비교할 때 앞에서와 같이 이들을 집합으로 보고, 집합간의 메트릭이 일정한 값  $a$  이내의 차이를 보이면, 이를 유사하다고 한다.

비교하고자 하는 두개의 시퀀스의 주기가 같은 경우와 다른 경우가 존재한다 만일, 주기가 같을 경우, 이의 순열을 앞에서 제시한 방법으로 비교할 수 있다. 그러나, 주기가 다른 경우, 벡터의 크기가 달라지므로 - (minus) 연산을 수행할 수 없다. 이 경우, 시퀀스내 존재하는 모든 원소에 대해서는 두 시퀀스의 충분히 많은 수의 원소가 일치하면서, 동일한 순서를 유지하고 있다면 이를 유사 검색의 후보에 포함시켜야 한다 이를 위해, 순열 검색은 대응하는 원소가 부족한 순열과도 검색을 수행할 수 있도록 수정되어야 한다.

원 시퀀스 (0 1 2 3 4 5 6 0 1 2 3 4 5 6) 분석된 주기 시퀀스 T=7, S1=(0,1,2,3,4,5,6)
---

주어진 시퀀스(순환일치성, 동일주기) T=7, S2=(0,1,2,3,4,5,6)  1. res = S2-S1 mod T 2. res = (0,0,0,0,0,0,0) 3. Accept S2 as a candidate
--

주어진 시퀀스(순환일치성, 동일주기) T=7, S2=(3,4,5,6,0,1,2)  1. res = S2-S1 mod T 2. res = (3,3,3,3,3,3,3) 3. Accept S2 as a candidate
--

주어진 시퀀스(순환일치성, 다른주기) T=5, S2=(1,2,3,4,5)  1. S2를 확장 S2=(1,2,3,4,5,m,m) 2. res = S2-S1 mod T 3. res = (0,0,0,0,0,u,u) 4. Accept S2 as a candidate
--

주어진 시퀀스(순환일치성, 다른주기) T=5, S2=(5,6,0,1,2)  1. S2를 확장 S2=(5,6,0,1,2,m,m) 2. res = S2-S1 mod T 3. res = (5,5,5,5,u,u) 4. Accept S2 as a candidate
--

그림 7 4가지 경우의 주기 시퀀스의 비교

이 과정은 그림 7에 설명되어 있다. 먼저 크기가 작은 시퀀스는 크기가 큰 시퀀스와 크기가 같아지도록 확장되고, 이 때 확장된 곳에는 원소 없음(m) 표시를 하여, 다음 단계에서 메트릭을 구할때 참고할 수 있도록 한다.

1) 본 논문에서는 실험을위해  $L_p$  메트릭,  $L_p = (\sum |x_i - y_i|^p)^{\frac{1}{p}}$ 를 이용하였는데[12],  $L_p$  메트릭은 유클리드 거리  $d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$ 를 n차원으로 확장한 일반식이다.

순열 비교의 결과는 일치 시 단위 벡터의 상수 배이다. 이 벡터는 모든 원소가 같은 값을 가지는 벡터이므로, 만일 다른 값을 가진다면 이 값으로부터 다른 정도만큼 떨어진 값을 선택하면 된다.

이를 유사 검색 매트릭으로 이용하기 위해, 단위 벡터의 각 원소를 정규화한 후 원소들 간의 표준 편차(standard deviation)를 계산하였다. 이 때 순열 검색을 위해  $(v_1 - v_2) \bmod n$  연산을 수행할 때, 대응하는 원소가 없으면(원소없음 표시가 되어있다), 결과값 대신 대응 원소 없음(u) 표시를 한다. 이 결과 표준 편차를 계산할 때 우리는 비교에서 대응되지 못한 원소의 갯수를 알 수 있으며, 대응되지 못한 원소를 필요에 따라, 고려하지 않고 값을 구할 수 있게 된다. 여기서 우리는 두가지 매트릭을 제안하여 사용하였다. 첫번째는 전체 주기의 원소들 중에서 비교에 참여한 원소의 비율이다.

$$\text{참여율} = \frac{\text{주기} - \text{대응원소없음의갯수}}{\text{주기}}$$

다음은 두 벡터의 - 연산 결과에 대해 그 분포를 구하는 매트릭으로서, 두 시퀀스가 얼마나 순서가 일치하는지를 정량적으로 나타내준다.

$$SD^2 = \frac{\sum_{i=1}^n v_i^2}{n} - \bar{v}^2$$

결과 벡터 내에서 표준 편차에 대응하지 못하는 값이 미치는 영향은 없으며, 항상 비교 연산의 매트릭  $m_m$ 은, 대응하지 못하는 값에 정확히 대응되는 원소가 있다고 가정할 경우의 매트릭  $m_e$ 에 대해, 항상  $m_e > m_m$ 이 되므로, 이 시퀀스는 항상 후보로서 선택된다. 이러한 순열 검색을 이용하면, 유사한 시퀀스는 항상 후보로 선택되므로 잘못된 기각(false dismissal)을 내지 않음을 알 수 있다.

이제 임의의 주기 시퀀스에 대해, 유사성 검색은 다음과 같이 수행한다.

- Step 1 시퀀스를 구성하는 원소들의 일치 확인
1. 각 집합의 크기를 확인. 차가  $threshold_p$  보다 작으면 다음단계로 계속.
  2. 각 집합의  $L_p$  매트릭을 측정.
  3. 각 원소의 포함 관계를 확인.
- Step 2 순열 확인
- 1 주어진 시퀀스  $S_1$ 에 대해, 등차 수열로 이루어진 순서 인덱스를 시퀀스의 순서에 맞게 할당한다.
  - 2 또 다른 시퀀스  $S_2$ 에 대해,  $S_1$ 의 시퀀스 디렉토리를 이용하여  $S_2$ 의  $S_1$ 에 대한 순열을 구한다. 이때 대응하는 원

3. 참여율이  $threshold_m$ 보다 크면 다음 단계로 계속한다.
- 4  $v_{res} = v_1 - v_2 \bmod n$ 를 계산한다. 이 때 '원소 없음(m)'이 표시된 원소쌍의 연산 결과는 '대응 원소 없음(u)' 표시를 한다.
5. 결과가  $c \cdot v_u$ 인지 확인한다.
6. 만일 결과가 단위 벡터의 상수배가 아니면, 전체 원소를 정규화한 후, 이의 SD(표준 편차)를 구한다. 이때 u 표시가 되어 있으면 이는 표준 편차의 계산에서 제외한다. 표준 편차가 미리 정한 임계치  $threshold_u$ 보다 작으면,  $S_2$ 를 유사 일치의 후보로 선택한다. (단,  $c$ 는 상수,  $v_u$ 는 단위 벡터)

4.5 검색 방법

주기 시퀀스에 대해 완전 일치 질의와 유사 일치 질의를 수행하기 위해 이를 시퀀스 원소에 대한 일치 확인과 순열의 일치 두가지의 부분제로 분할하였는데, 각 단계별로 효율적인 검색을 위해 여러가지 매트릭과 인덱스 기법을 고려하였다. 완전일치검색과 유사검색을 지원하기 위해 다음과 같은 주기 시퀀스의 비교 방법을 사용하였다.

```
Algorithm 2 element_equivalence(string seq0,string seq1)
1 /* compare the size of seq0 and seq1 */
2 if the difference is less than thresholdp go next step
3 d = Lp(seq0, seq1) /* Lp는 Lp 매트릭함수 */
4 if d = 0
5   Choose seq1 as a candidate of a exact matched set
6 else if d < thresholdp
7   Choose seq1 as a candidate of a similar matched set
8 else
9   Discard seq1
```

```
Algorithm 3 permutation_equivalence( permutation ps,
permutation pt)
1 permutation t /* permutation은 하나의 순열을 저장하기 위한 타입 */
2 double sd
3 int size, ratio
4 build the permutation of pt for the permutation ps
5 size = length(ps)
6 t = (ps - pt) mod size
7 sd = standard - deviation (t)
8 ratio = length(pt) / length(ps)
9 if sd = 0
10 /* Two permutations are equal */
11 return TRUE
12 else if ratio < thresholdm
```



```

13      /* Missing elements are too many Two
           permutations are discarded. */
14      return FALSE
15  else if  $sd < threshold_u$ 
16      /* Two permutations are similar. */
17      return TRUE
18  else
19      /* Two permutations are different */
20      return 0
    
```

5. 구현 및 결과

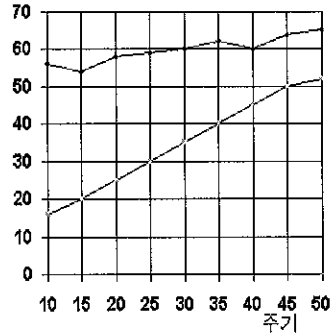
본 절에서는 주기 시퀀스 검색의 유용성을 보이기 위해, 여러가지 시퀀스를 생성시켜 이에 대한 검색 결과를 보였으며, 비디오 내용 검색과 관련한 검색 성능의 비교를 위해 [6]에서 제시한 검색 방법을 통해 얻어진 시퀀스 및 시퀀스의 비교 방법을 본 논문의 시퀀스의 비교 방법과 비교 하였다.

비교의 방법은 첫째, 일반적인 주기를 가지지 않는 시퀀스에 대해, 주기를 가지지 않는 패턴의 검색을 수행하여 보았다. 두번째로는 일반적인 주기를 가지지 않는 시퀀스에 대해, 주기를 가지는 패턴의 검색을 수행하여 보았다. 세번째로는 일반적인 주기를 가지는 시퀀스에 대해, 주기를 가지는 패턴의 검색을 수행하여 보았다. 여기에서 사용된 데이터는 여러 가지 동작을 촬영한 임의의 비디오로부터 분석한 결과 시퀀스 100개가 이용되었으며, 검색 성능의 비교는 검색 횟수를 기준으로 행하였다.

첫번째 경우, 주기를 가지지 않는 시퀀스에 대해, 주기 분석을 수행하게 되면, 주기는 전체 시퀀스의 길이가 얻어지게 되며, 한주기의 시퀀스는 전체 시퀀스가 되어 검색 시의 비교 횟수는 일반 시퀀스 검색과 동일하게 된다. 그러나 두번째와 세번째 경우에 있어서는 그림 8의 검색 결과와 같이 본 논문에서 제시한 검색 방법을 주기를 가지는 시퀀스에 대해, 주기를 가지는 패턴을 이용하여 검색할 때, 그 성능이 매우 좋을을 보여주고 있다. 즉 원 시퀀스가 주기를 가지는 것을 예측할 수 있을 때 이를 미리 분석하여 두면, 검색 시에 매우 빠르게 시퀀스를 검색해 낼 수 있음을 알 수 있다. 그림 8의 두번째 그래프에서는 검색을 위해 검색 대상 데이터베이스 내에 포함된 시퀀스의 주기를 50으로 가정하였으므로 원 시퀀스의 길이가 50을 넘게 되면, 본 논문에서 제시된 방법을 이용하여 비교할 때, 비교하고자 하는 대상 시퀀스의 길이가 50이므로 비교 횟수는 원 시퀀스의 길이가 50일 때와 동일하다. 즉, 원 시퀀스의 길이가 데이터베이스 내에 포함된 데이터의 주기보다 커지면,

기존의 시퀀스 검색과 성능이 같아지며 그 외의 경우에는 본 논문에서 제시한 방법의 검색횟수가 작다.

시퀀스의 길이 고정시 비교횟수



시퀀스의 주기 고정시 비교횟수

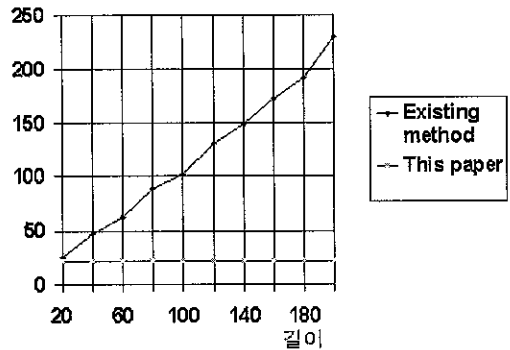


그림 8 주기시퀀스의 검색 시 비교 횟수

본 논문에서는 이러한 검색 방법을 이용하여 비디오에 포함된 동작 데이터를 검색해보았다. 비디오데이터는 약 100 프레임 정도로 구성되는 서로 다른 동작을 나타내는 비디오 조각 13개를 준비하고, 여기에 포함된 동작을 검색하기 위해, 비디오로부터 대상물의 외곽선을 추출하고, 여기서 특징 벡터를 단순화하기 위하여 MBR(Minimum Bounding Rectangle)을 추출하였다.<sup>2)</sup> 여기서 평행 이동이나 크기에 관한 정규화를 통해 불변속성(invariant)를 추출하였다. 이를 위해 검색에 사용된 디지털 비디오 데이터는 제한적인 동작만을 담고 있다. 검색에 사용될 데이터를 수집하기 위해 카메라를 고정

2) 여기서 검색의 대상을 단순화하기 위해 MBR을 가정하였을 뿐, [11]등에서 제시된 방법으로 특징 벡터와 이의 시퀀스를 추출할 경우에도 동일한 결과를 얻을 수 있다.

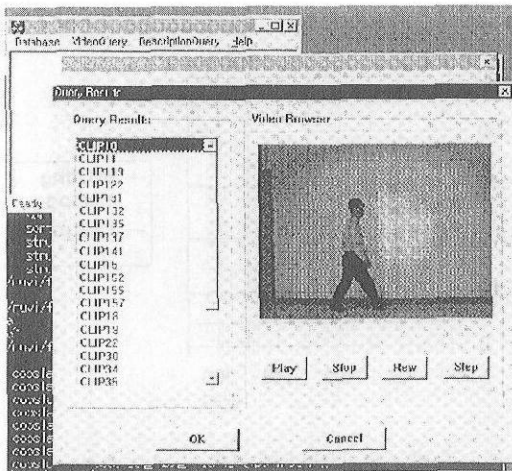
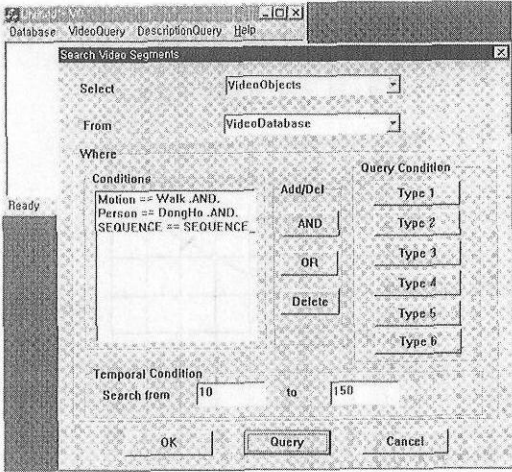


그림 9 동작 예의 제시를 이용한 내용 기반 동작 검색

시킨 후 카메라에 수직 방향으로 진행하거나 정지상태로 반복이 있는 동작들을 수행하게 한 후, 이를 비디오로 녹화하였다. 이를 Sun과 SunVideo MPEG 장비를 이용하여 MPEG1 화일로 변환하였다. 변환된 MPEG 화일로 부터 일정한 간격으로 프레임 추출한 후, 각 프레임 화면에 대해, 외곽선을 추출한 후, 이로부터 몇개의 특징 점을 뽑아, 정규화하여 대상물의 불변 속성(invariant)으로 사용할 수 있게 하였다. 여기서 추출된 불변 속성들을 DBMS에서 처리할 수 있도록 그림 5의 자료 구조에 따라 동일한 스키마를 가지는 레코드들의 집합으로 변환하고, 본 논문에서 제시한 특징 시퀀스로 표현하여 제시한 검색 방법을 이용하여 검색을 수행하여 보았다.

추출된 불변 속성들은 동작 데이터이므로 주기를 가진다는 사실을 미리 예측할 수 있다. 그러므로 이 데이터들은 주기 시퀀스 모델을 이용하여 표현될 수 있다. 우리는 총 13가지의 동작에 대한 비디오 데이터를 대상으로 특징을 추출하고 시퀀스를 구성해보았다. 질의 예에 의한 질의 방식(Query By Example)로 수행하였는데, 추출된 주기 시퀀스에 대해 각각 임의의 레이블을 붙이고 이를 질의에 이용하는 방식을 취하였다. 이러한 레이블을 편의 상 동작과 관련있는 단어를 이용하였으며, 질의 시에 이를 사용하였다. 구성된 시퀀스에는 시퀀스가 표현하는 동작을 레이블로 붙여 검색의 조건으로 레이블을 사용할 수 있게 하였다. 그림 9는 질의 예로 미리 분류된 동작을 이용하는 내용 기반 동작 검색의 질의 윈도우이다.

### 6. 요약 및 향후 연구

우리는 본 논문에서 시퀀스 데이터가 주기를 가지는 경우가 많으며, 이 경우 주기는 데이터의 가공 초기에 쉽게 얻어질 수 있다는 점에 착안하였다. 이러한 성질을 시퀀스의 검색에 이용하기 위해 주기 시퀀스 모델을 제시하였다. 주기 시퀀스의 검색을 위해 등차 수열로 이루어진 순열의 인덱스에 대해서는 이를 순환시키는 순열을 적용하였을 때, 원 순열과 순환된 순열과의 차가 일정한 값으로 얻어진다는 순환 순열의 성질을 보였다.

이러한 성질을 이용하여, 주기 시퀀스로 표현된 시퀀스 데이터를 효율적으로 검색할 수 있는 방법을 제시하였다. 검색 방법으로는 주어진 주기 시퀀스에 대해 완전 일치 검색 및 유사 검색 방법을 제시하였다. 이러한 검색 방법은 주기를 가지는 원 시퀀스에 대해, 주기를 가지는 패턴을 검색할 경우, 주기 정보를 사용하지 않은 검색에 비해 매우 효율적임을 보였다. 또한 검색 대상에 대해, 검색의 결과가 될 수 있는 시퀀스를 실수로 선택하지 않는 잘못된 기각을 하지 않음을 보였다.

우리의 검색 방법은 (1) 시퀀스의 원소들의 일치 검색 (2) 순열 검색의 두 단계로 나뉘어져 있으며, 각 단계 별로, 완전 일치 검색과 유사 일치 검색을 고려하여, 검색 방법을 제시하였다. 각 주기 시퀀스들을 비교하기 위해서  $L_p$  메트릭을 이용하였으며, 순열 검색을 위해서는 벡터에 대한 - 연산 및 mod 연산만을 사용하여 효율적인 검색이 되도록 하였다.

우리는 제안한 방법을 적용하여 멀티미디어 스트림 데이터인 비디오 데이터로부터 불변 속성을 추출하여 이를 주기 시퀀스로 표현한 후, 주기 시퀀스 검색을 이

용하여 동작 간의 검색을 수행하여 보았으며, 그 검색 결과를 보였다. 본 논문에서는 시퀀스 데이터가 주기를 가진다는 것을 가정하였고 이를 데이터의 가공 초기에 쉽게 인식할 수 있음을 가정하였다. 그러나 이 가정은 모든 일반적인 시퀀스나 멀티미디어 스트림에 대해서는 주기 검색에 대한 비용 증가의 단점으로 효율이 저하될 수도 있다. 단점이 있다. 그러므로 이를 겉으로 주기가 드러나지 않는 시퀀스에 대해서 효율적으로 적용할 수 있도록 개선하는 연구가 필요하다.

### 참 고 문 헌

- [1] Sibel Adali, K. Selcuk Candan, Su-Shing Chen, Kutluhan Erol, V. S. Subrahmanian, "The Advanced Video Information System: data structures and query processing", *Multimedia Systems*, 4, 1996.
- [2] Rakesh Agrawal, Christos Faloutsos, Arun Swami, "Efficient Similarity Search In Sequence Databases", Technical report, IBM Almaden Research Center, Harry Road, San Jose CA 95120, 1993. Also in Lecture Notes in Computer Science 730, Springer Verlag, 1993, 69-84.
- [3] Rakesh Agrawal, King-Ip Lin, Harpreet S. Sawhney, Kyusok Shim, "Fast Similarity Search in the Presence of Noise, Scaling, and Translation in Time-Series Databases", In *Proceedings of 21st International Conference on VLDB*, September 1995.
- [4] Tat-Seng Chua, Swee-Kiew Lim, Hung-Keng Pung, "Content-based Retrieval of Segmented Images", In *Proceedings of Second ACM International Conference on Multimedia*, 211-218, October 1994.
- [5] Young Francis Day, Serhan Dagtas, Mitsutoshi Iino, "Object-Oriented Conceptual Modeling of Video Data", In *Proceedings of Second ACM International Conference on Data Engineering*, 401-408, 1995.
- [6] Nevenka Dimitrova, Forouzan Golshani, "Rx for Semantic Video Database Retrieval", In *Proceedings of Second ACM International Conference on Multimedia*, 219-216, October 1994.
- [7] Christos Faloutsos, King-Ip(David) Lin, "FastMap: A Fast Algorithm for Indexing, Data-Mining and Visualization of Traditional And Multimedia Datasets", In *Proceedings of the 1995 ACM SIGMOD*, 163-174, May 1995.
- [8] Christos Faloutsos, M. Ranganathan, Yannis Manolopoulos, "Fast Subsequence Matching in Time-Series Databases", In *Proceedings of the 1994 ACM SIGMOD*, 419-429, May 1994.
- [9] Michael R. Garey, David S. Johnson, *Computers and Intractability*, W. H. Freeman and Company, 1979.
- [10] Venkat N. Gudivada, Vijay V. Raghavan, "Design and Evaluation of Algorithms for Image Retrieval by Spatial Similarity", *ACM Transactions on Information Systems*, 13(2):115-144, April 1995.
- [11] M. K. Leung, Y. -H. Yang, "First Sight: A Human Body Outline Labeling System", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(4):359-377, April 1995.
- [12] King-Ip Lin, H. V. Jagadish, Christos Faloutsos, "The TV-tree - an index structure for high-dimensional data", *VLDB Journal*, 4, 1995.
- [13] Maveed U Qazi, Miae Woo, Arif Ghafoor, "A Synchronization and Communication Model for Distributed Multimedia Objects, In *Proceedings of ACM Multimedia '93*, August 1993.
- [14] Robert Sedgwick, *Algorithms in C++*, Addison Wesley Publishing Company, 4th edition, 1989.
- [15] Steven M. Seitz, Charles R. Dyer, "Affine Invariant Detection of Periodic Motion", Technical report 1225, University of Wisconsin Madison, WI 53706, June 1994.
- [16] K. Siddiqi, B. B. Kimia, "Parts of Visual Form: Computer Aspects", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(3):239-251, March 1995.
- [17] Ron Weiss, Andrzej Duda, "Composition and Searching with a Video Algebra", *IEEE Multimedia*, 2(1), 1995.



김 기 병

1990년 2월 서울대학교 자연과학대학 계산통계학과 이학사. 1992년 2월 서울대학교 자연과학대학 계산통계학과 전산과학전공 이학석사. 1991년 ~ 1992년 서울대학교 중앙교육연구전산원 조교. 1994년 2월 서울대학교 공과대학 컴퓨터공학과 박사과정 수료. 1992년 ~ 서울대학교 공과대학 컴퓨터공학과 박사과정, 1997년 ~ 현재 한국 휴렛팩커드(주) 컨설팅사업본부 전자상거래팀 과장. 관심분야는 데이터베이스, 멀티미디어 데이터베이스, 이미지/비디오 검색, 전자상거래.



김 명 원

1972년 서울대학교 응용수학과 졸업(학사). 1981년 Univeristy of Massachusetts(Amherst) Computer Science 석사학위 취득. 1986년 University of Teax(Austin) Computer Science 박사학위 취득. 1975년 ~ 1978년 한국과학기술연구소 연구원. 1982년 ~ 1985년 Institute for Computing Science & Cimputer Applications(Univ. of Texas) 연구원. 1985년 ~ 1987년 AT&T Bell Labs. (Naperville) Member of Technical Staff. 1987년 ~ 1994년 한국전자통신연구소 책임연구원. 1991년 ~ 1993년 충남대학교 전자계산학과 겸임 부교수. 1994년 ~ 현재 숭실대학교 컴퓨터학부 부교수. 1992년 ~ 1993년 한국신경회로망연구회 회장. 1993년 ~ 1995년 정보과학회 뉴로컴퓨팅 연구회 위원장. 1993년 ~ 현재 IEEE Neural Network Council 한국지부장. 관심분야는 유연추론, 신경회로망, 퍼지 시스템, 진화알고리즘, 패턴인식, 자동추론, 기계학습, creativity engineering 등.



김 형 주

1982년 2월 서울대학교 컴퓨터공학과 졸업. 1985년 8월 Univ. of Texas at Austin, 전자계산학 석사. 1988년 5월 Univ. of Texas at Austin, 전자계산학 박사. 1988년 5월 ~ 9월 Univ. of Texas at Austin. Post-Doc. 1988년 9월 ~ 1990년 12월 Georgia Institute of Technology, 부교수. 1991년 1월 ~ 현재 서울대학교 컴퓨터공학과 부교수. 관심분야는 객체지향 시스템, 사용자 인터페이스, 데이터베이스.